US 20030074377A1

(19) United States

(12) Patent Application Publication (10) Pub. No.: US 2003/0074377 A1

Gupta et al. (43) Pub. Date: Apr. 17, 2003

(54) METHOD AND APPARATUS FOR REPORTING THE STATUS OF WORK IN PROGRESS

(76) Inventors: Aman Gupta, Milwaukee, WI (US); Amit Maheshwari, Brookline, MA (US); James A. Yenerich, Pewaukee, WI (US)

Correspondence Address:
ZIOLKOWSKI PATENT SOLUTIONS GROUP, LLC (GEMS)
14135 NORTH CEDARBURG ROAD
MEQUON, WI 53097 (US)

(21) Appl. No.: 09/747,647

(22) Filed: Dec. 22, 2000

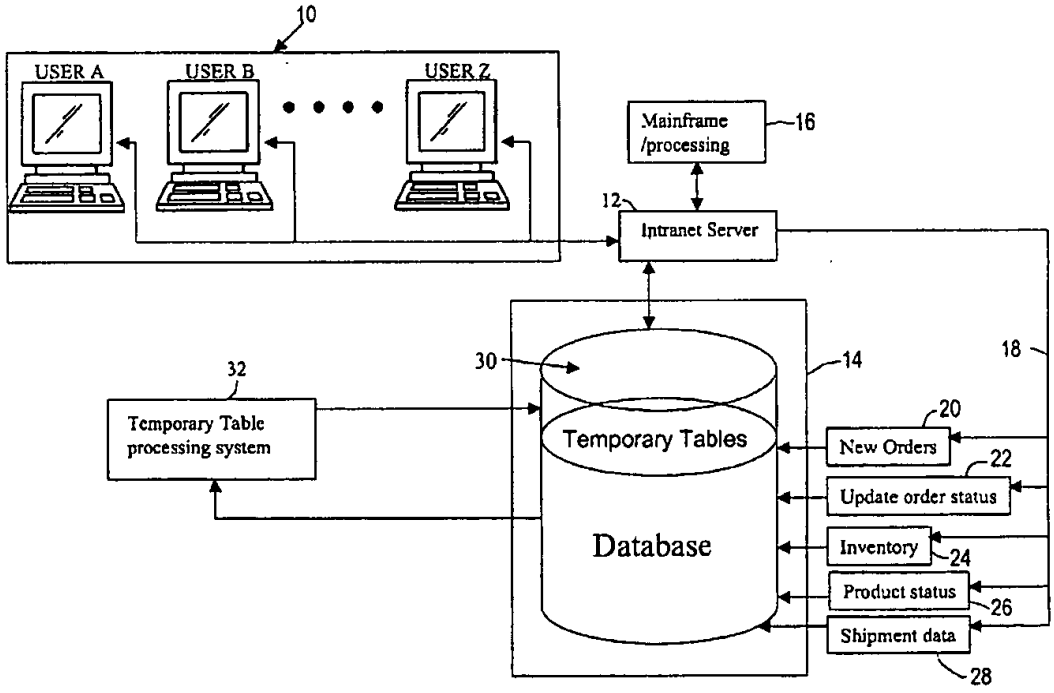
Publication Classification

(51) Int. Cl.[7] G06F 12/00

(52) U.S. Cl. 707/203

(57) ABSTRACT

A system and method are disclosed for electronically reporting in real-time the status of work in progress by a computerized system. The invention provides alert notification when a customer requested shipment date falls before the date that a shipment was promised or is scheduled to be fulfilled. The invention further includes notification when the request date for shipment is near, thereby providing a highlighted alert notification to managers. The system is designed to update its data when a request is made, or automatically in real-time or on a regular basis. A proactive alert is provided to alert managers of a correctable problem for that order and a reactive alert is provided to track late shipments to prevent future late shipments.



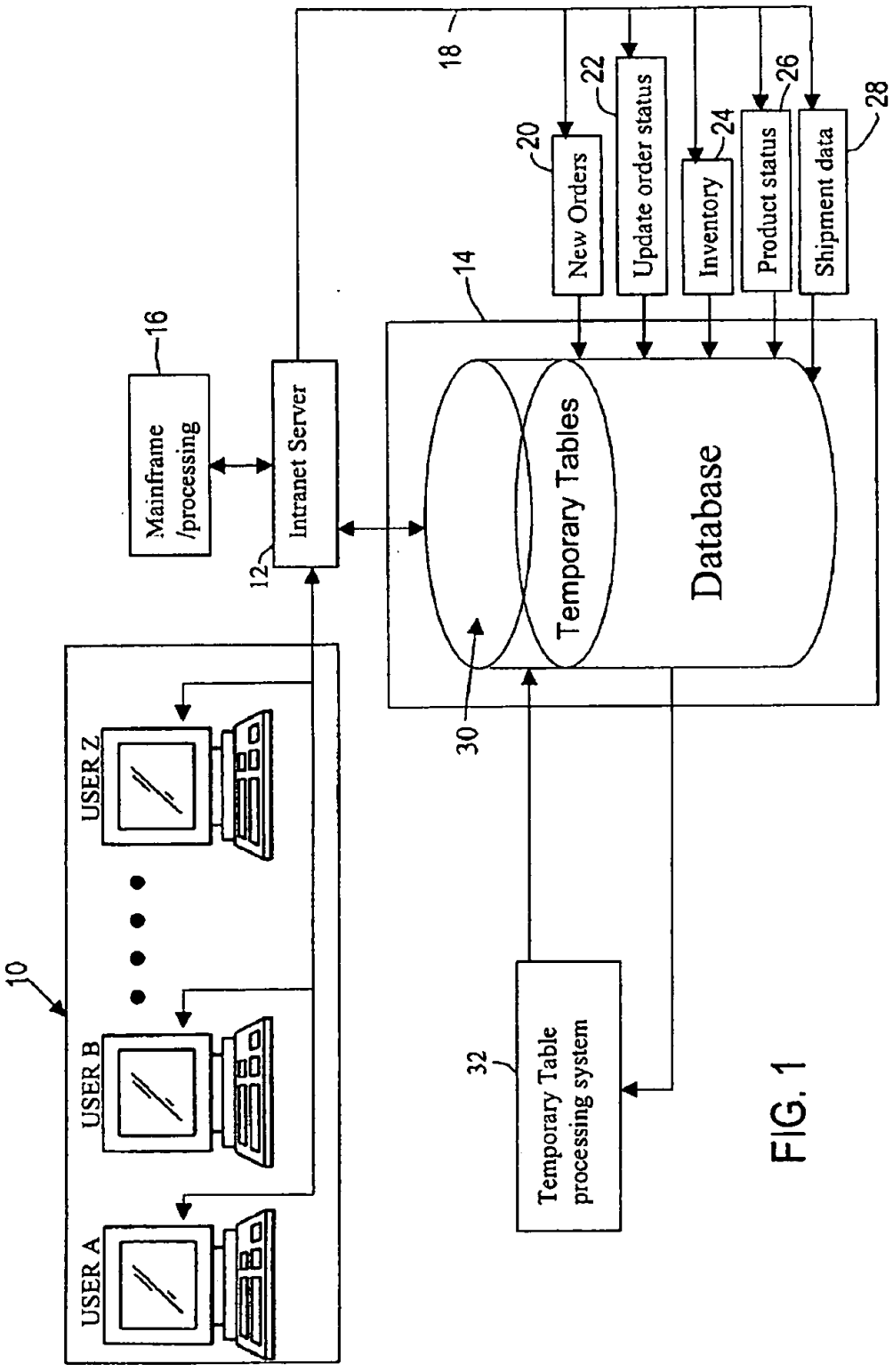

10
USER A
USER B
USER Z
12
Intranet Server
16
Mainframe /processing
18
14
Database
Temporary Tables
30
20
New Orders
22
Update order status
24
Inventory
26
Product status
28
Shipment data
32
Temporary Table processing system

FIG. 1

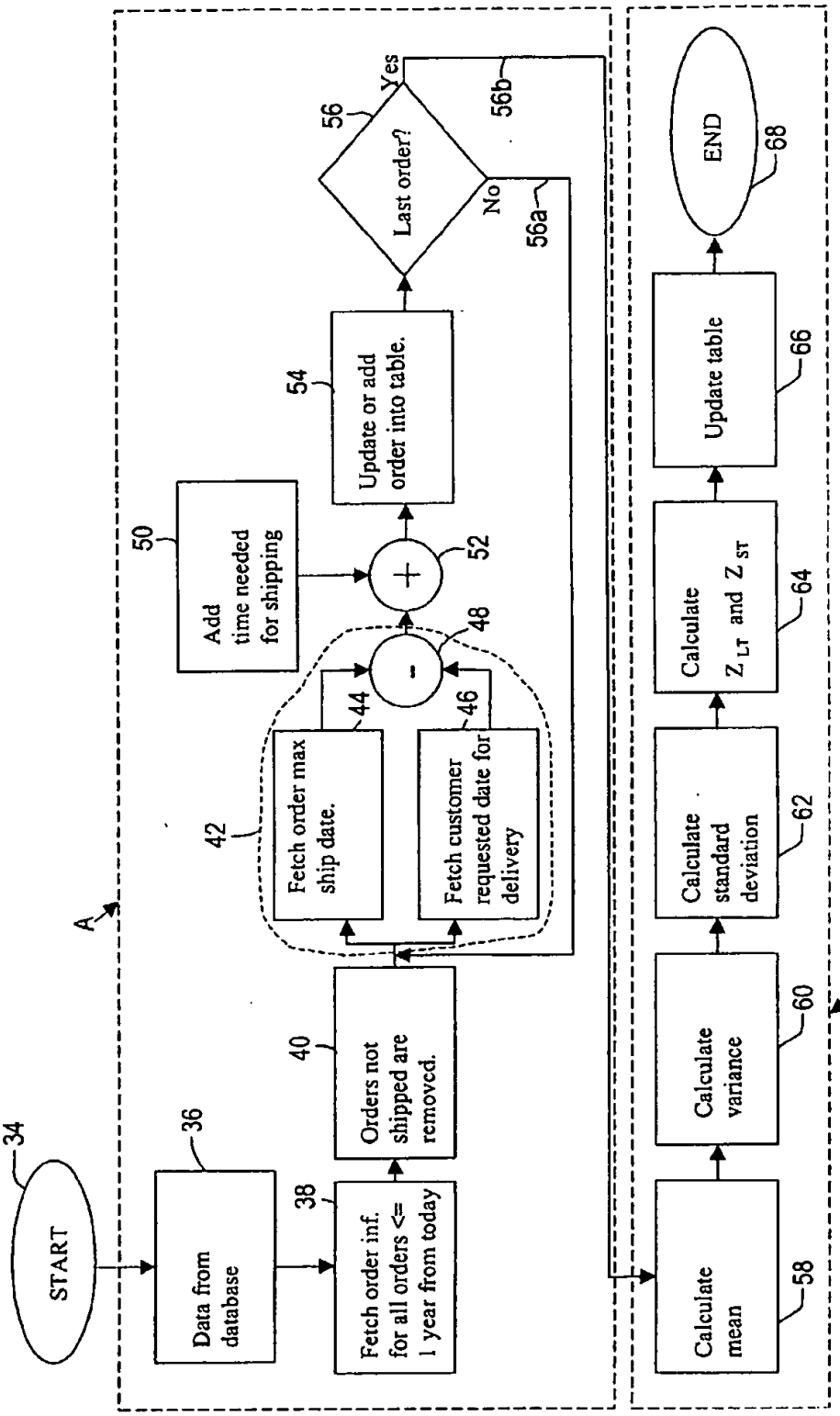

START
34
Data from database
36
Fetch order inf. for all orders <= 1 year from today
38
Orders not shipped are removed.
40
A
42
Fetch order max ship date.
44
Fetch customer requested date for delivery
46
-
48
+
52
Add time needed for shipping
50
Update or add order into table.
54
Last order?
56
Yes
56b
No
56a
Calculate mean
58
Calculate variance
60
Calculate standard deviation
62
Calculate Z_{LT} and Z_{ST}
64
Update table
66
END
68
B

FIG. 2

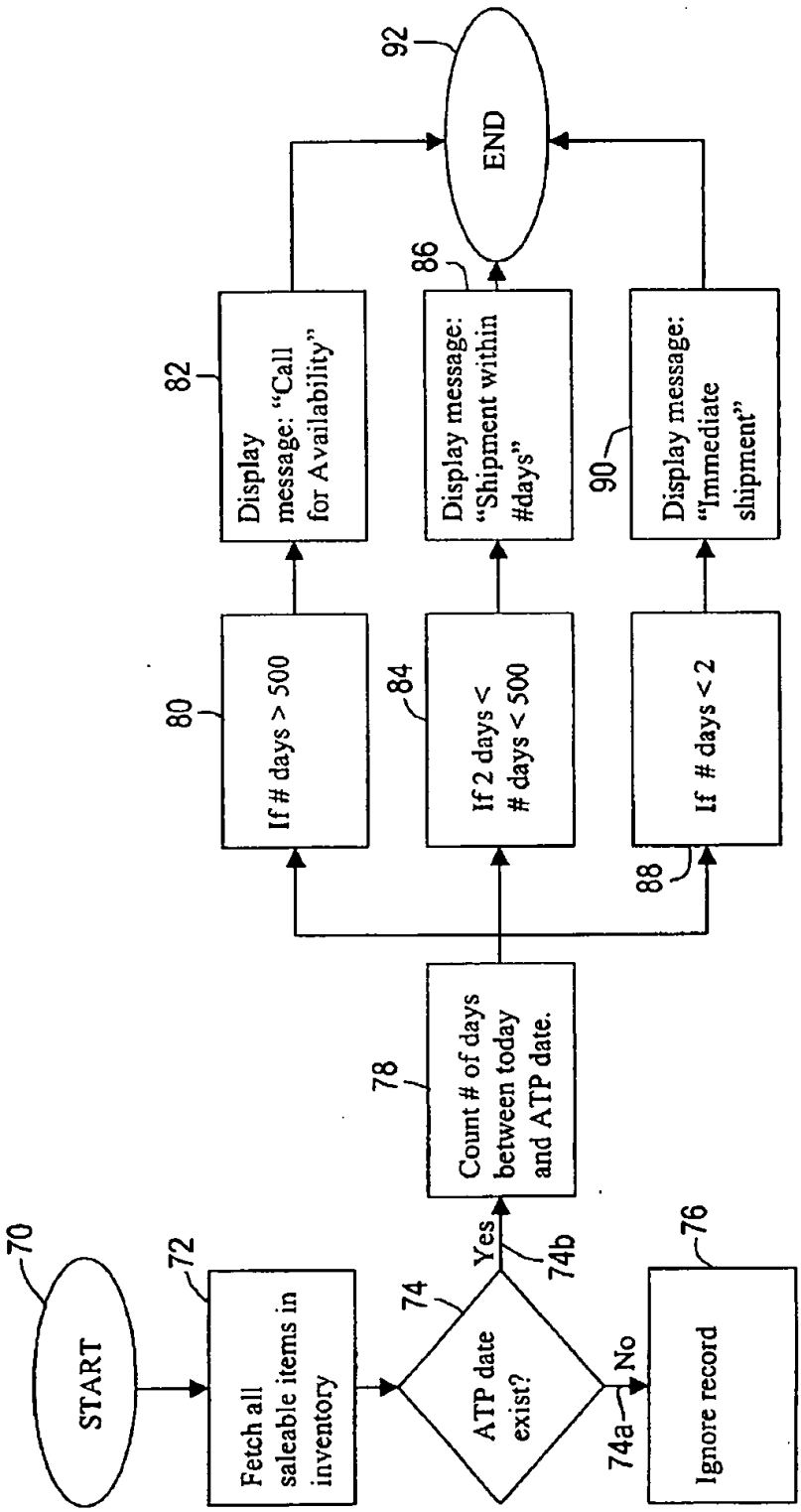

START
70
Fetch all saleable items in inventory
72
ATP date exist?
74
Yes
74b
No
74a
Ignore record
76
Count # of days between today and ATP date.
78
If # days > 500
80
If 2 days < # days < 500
84
If # days < 2
88
Display message: "Call for Availability"
82
Display message: "Shipment within #days"
86
Display message: "Immediate shipment"
90
END
92

FIG. 3

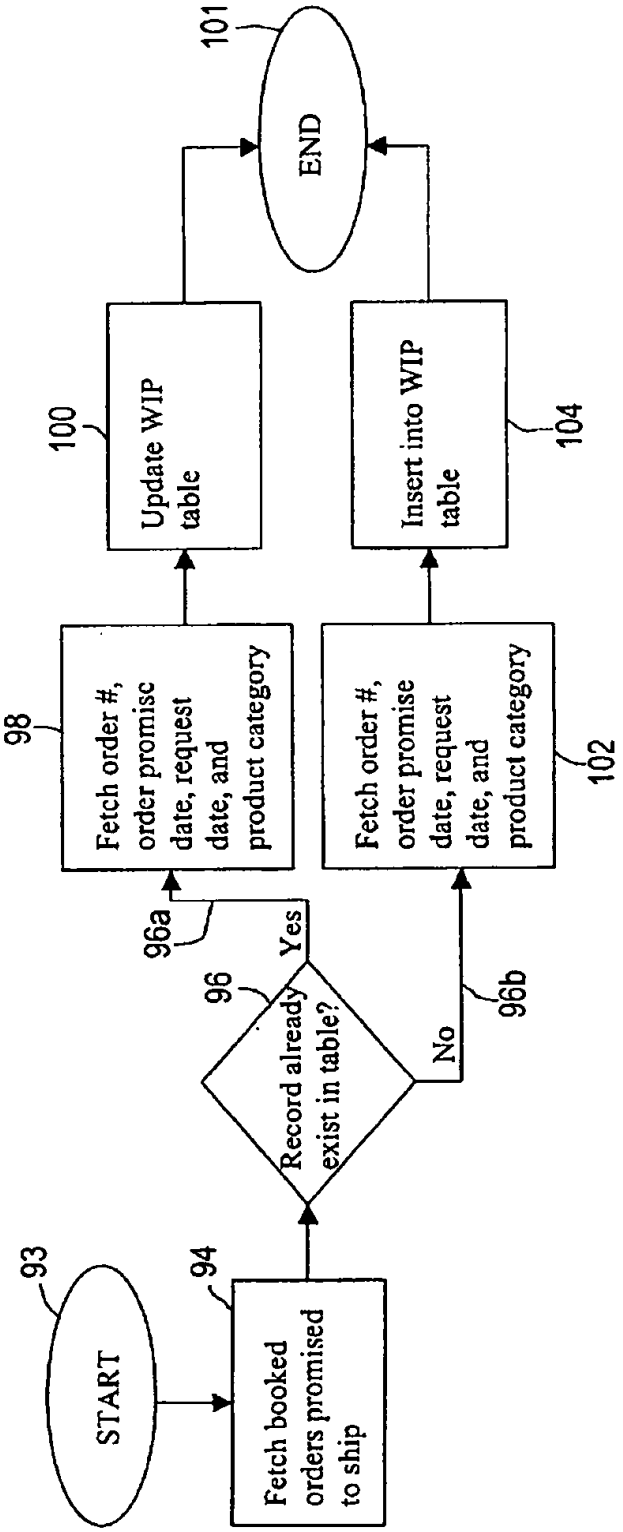

START
93
Fetch booked orders promised to ship
94
Record already exist in table?
96
Yes
96a
No
96b
Fetch order #, order promise date, request date, and product category
98
Fetch order #, order promise date, request date, and product category
102
Update WIP table
100
Insert into WIP table
104
END
101

FIG. 4

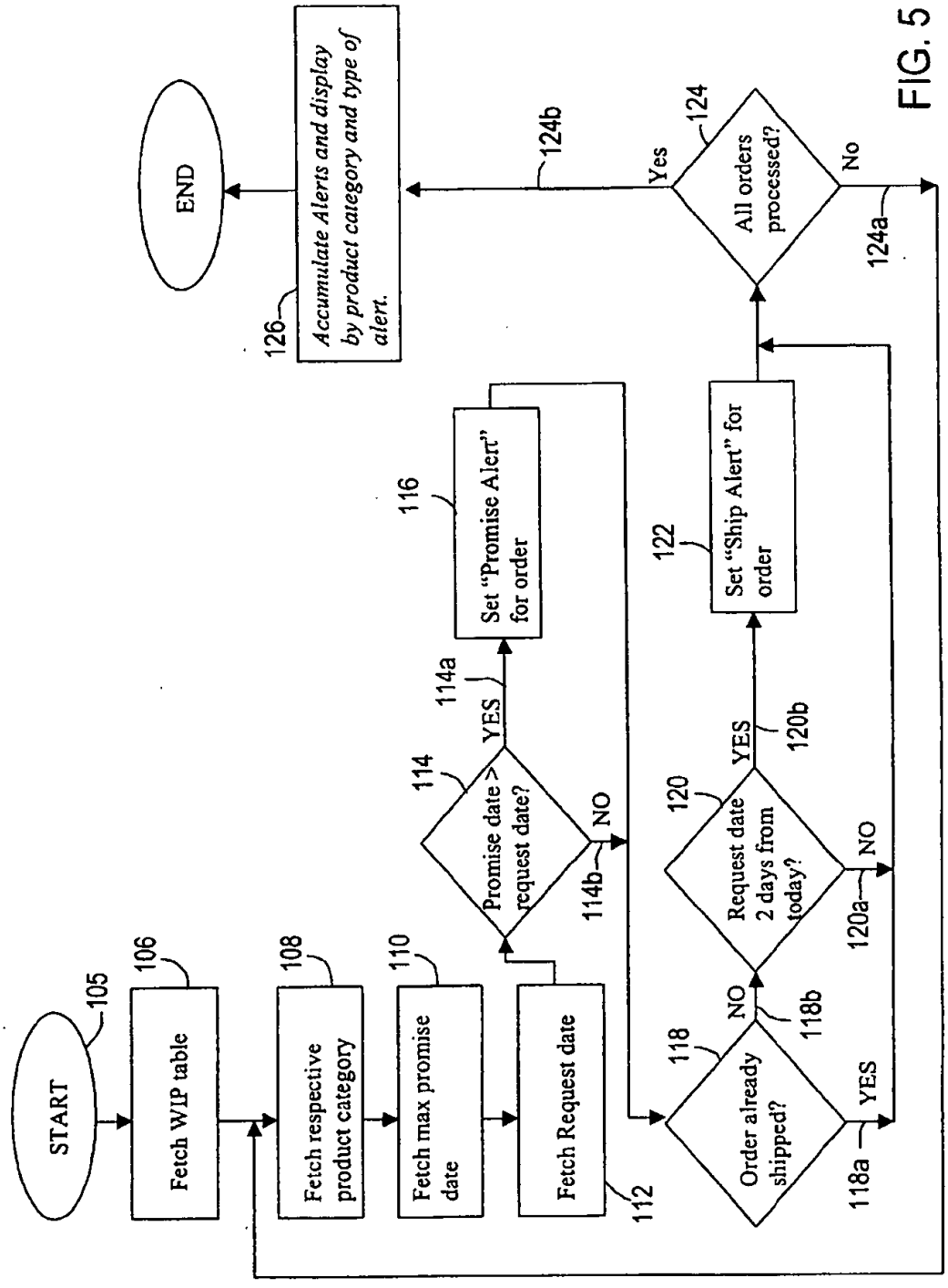

START
105
Fetch WIP table
106
Fetch respective product category
108
Fetch max promise date
110
Fetch Request date
112
Promise date > request date?
114
YES
114a
NO
114b
Set "Promise Alert" for order
116
Request date 2 days from today?
120
YES
120b
NO
120a
Set "Ship Alert" for order
122
Order already shipped?
118
NO
118b
YES
118a
All orders processed?
124
Yes
124b
No
124a
Accumulate Alerts and display by product category and type of alert.
126
END

FIG. 5

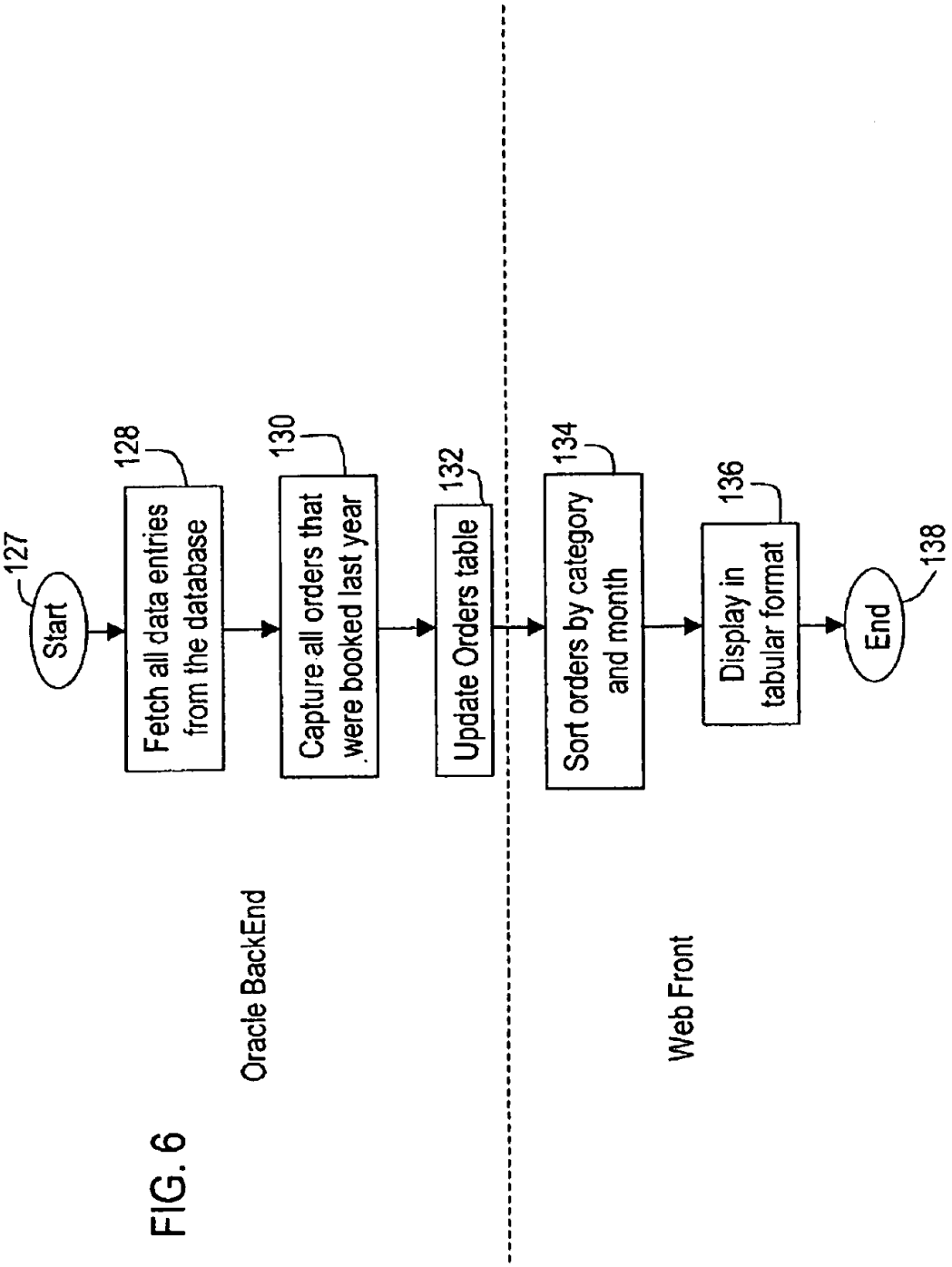
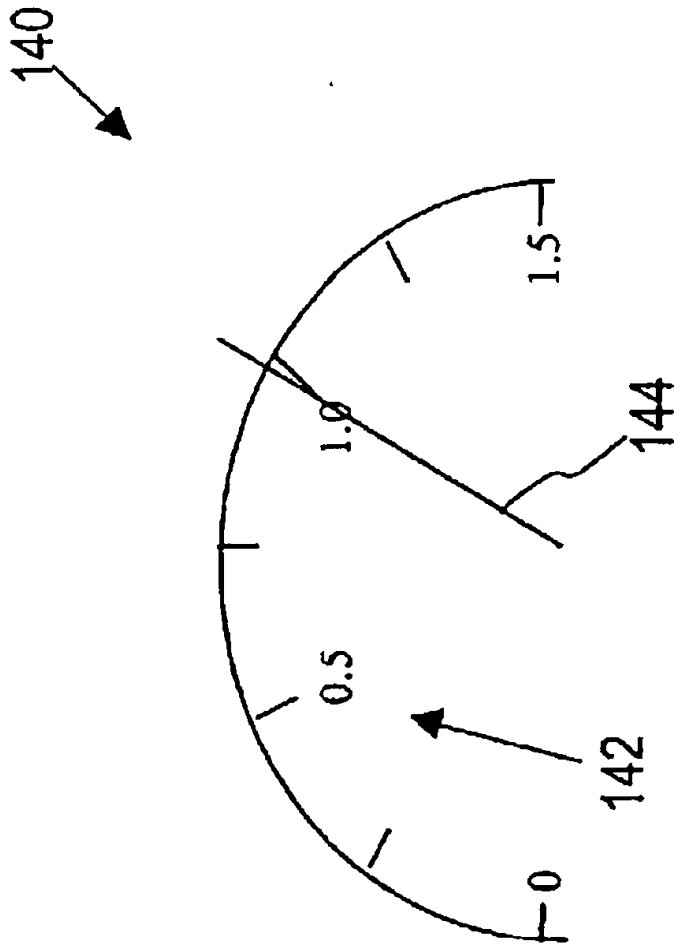

127
Start
128
Fetch all data entries from the database
130
Capture all orders that were booked last year
132
Update Orders table
134
Sort orders by category and month
136
Display in tabular format
138
End
Oracle BackEnd
Web Front

FIG. 6

FIG. 7



METHOD AND APPARATUS FOR REPORTING THE STATUS OF WORK IN PROGRESS

BACKGROUND OF THE INVENTION

[0001] The present invention relates generally to electronic monitoring of the status of work in progress and more particularly to electronically reporting in real-time shipping readiness of a product being produced.

[0002] In any business, delivery of product or services within the time expectations of the customer is necessary to be successful. Customers expect delivery of a product or service on or before the date delivery was requested and if the business is consistently not meeting customers' expectations, they risk losing the client to a competitor and a reduction in market share. For many products or services, if delivery is earlier than expected, the customer is happy as most customers appreciate delivery of goods or services at any time before the expected date of delivery. However, some products by their very nature i.e., perishable goods, need to be delivered on a specific date or within a very narrow time span defined by the customer.

[0003] For example, for some products, a customer may need to prepare a space for installation of the to-be-delivered product and, as such, may require a specified amount of lead-time. Thus if the product is delivered too early, it may cause additional costs to the customer for storage or other requirements. Additionally, if the shipment is late, the customer may begin to suffer a loss for every hour or day that the newly prepared site sits empty, waiting for the arrival of the product to install.

[0004] It is therefore desirable to reduce variation in shipping dates to within a defined span of time near or at the customer's requested date for shipment. An automated communication-reporting tool for managers trying to reduce variation in shipping dates would be highly desirable to support this task.

SUMMARY OF THE INVENTION

[0005] The present invention discloses a method and system for electronically reporting real-time status of work in progress. An alert system is disclosed that contains both proactive and reactive alerts. The alert scheme is based upon the electronically warning of mismatched future promised delivery dates and customer requested shipping dates. The alerts assist with avoiding or greatly reducing any problems with meeting customer expectations.

[0006] In accordance with one aspect of the invention, a method for reporting status of work in progress is disclosed wherein a database is periodically queried for information about product orders. The information obtained for each order includes: an order number, a promise date, a request date, a shipment date, and a product category for a plurality of products/services offered. The method further provides for comparing the promise dates and the request dates. When compared, it is decided whether an alert should be set, including setting a proactive promise alert if a promise date is later than a request date for a given order. The following step of this method includes displaying the proactive promise alerts with the order number, product category, and type of alert.

[0007] In accordance with another aspect of the invention, a computer-readable medium is disclosed, having stored thereon one or more computer programs. The computer program(s), when executed by one or more computers, causes the one or more computers to display electronic shipment alerts. The program begins by instructing the one or more computers to populate a database with data to include an order number, a promise date, a request date, a shipment date, and a product category for a plurality of orders. The one or more computers are additionally instructed to periodically query the database and compare promise dates to request dates. When compared, the one or more computers are instructed to set a proactive alert if the promise date is later than a request date, and set a reactive alert if the shipment date exists and the request date is less than a user-defined number of days prior to a current date. The one or more computers are then instructed to display any promise and shipment alerts by product category and type of alert.

[0008] In accordance with yet another aspect of the invention, a computer data signal is disclosed, representing a sequence of instructions, that when executed by one of more processors, causes the one or more processors to display proactive and reactive alerts by product/service category and type of alert. The sequence of instructions first causes the one or more processors to populate a database with an order date indicating a date an order is initially made, a request date indicating a date when a customer requests delivery of the order, a shipment date, when available, indicating a date when actual shipment will occur, and a product/service category for each order for a product/service. The one or more processors are further instructed to query the database and compare promise dates to request dates for each order, and to check for the entry of a shipment date for each order. The one or more processors are then instructed to set a proactive alert if any promise date is later than a request date, and to set a reactive alert if a shipment date exists for an order and the request date is less than a user-defined number of days prior to a current date. The one or more processes are lastly instructed to display all proactive and reactive alerts by product/service category and type of alert.

[0009] Various other features, objects, and advantages of the present invention are made apparent by the following detailed description and drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

[0010] The drawings illustrate one preferred embodiment presently contemplated for carrying out the invention.

[0011] In the drawings:

[0012] **FIG. 1** is a high-level overview block diagram representing an embodiment of the invention.

[0013] **FIG. 2** is a flow chart representation of a process to display quality indicators in accordance with one aspect of the invention.

[0014] **FIG. 3** is a flow chart representation of the availability reporting process in accordance with another aspect of the invention.

[0015] **FIG. 4** is a flow chart representation of the work in progress table update process in accordance with another aspect of the invention.

[0016] **FIG. 5** is a flow chart representation of the shipment and promise alert setting and display process in accordance with another aspect of the invention.

[0017] **FIG. 6** is a flow chart representation of the process to display orders and revenue in accordance with another aspect of the invention.

[0018] **FIG. 7** is a representation of the Z-value graphic indicator in accordance with another aspect of the invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

[0019] In a preferred embodiment of the present invention, a program is designed to augment a commercially available database program. Specifically, it is designed as a separate program that works with data in the database. The program provides real-time, or near real-time reporting capabilities that are not available to users of the commercially available database. Other embodiments of the invention may use different hardware and/or software manifestations to embody the invention in accordance with their particular design.

[0020] Referring to **FIG. 1**, an overview diagram of a reporting system is shown which includes a plurality of user stations **10** such as User A, User B through User Z. These user stations **10** represent any number of users on a network at any time. The user stations **10** are connected to an Intranet server **12**. This can be through direct communication, a local area network, or from another network like an intranet or the Internet. It may also include user stations **10** connected via a variety of other networking connections, including wireless connections. The Internet server **12** is in communication with database **14**, which may be on another computer network. The Intranet server **12** is also communicatively connected to a mainframe/processing section **16**. The Mainframe/Processing section **16** processes data from database **14** based on user requests and completes various reports and provides the reports to the user stations **10**.

[0021] Database **14** contains data on orders, availability (when products will be available for shipping), requested shipping dates, actual shipping dates, promised shipping dates, revenue per order, and various other product and sales information. All of the information is constantly updated at the user stations **10**, and is communicated through the Intranet server **12**. Some of the users may update the information and some may merely be accessing the database for information. The boxes **20-28** in **FIG. 1** show the input of this information into the database **14**, indicated by data input **18**. The data includes new orders **20**, update order status **22**, current inventory **24**, product status **26**, and shipment data **28**.

[0022] The database **14** has the ability to reserve areas for computational results known as temporary tables **30**. The tables contain the latest totals programmed to be stored therein that can be readily accessed by the users at user stations **10**. It is temporary in that the temporary table processing system **32** repeatedly updates the temporary tables **30**. When the data in the database **14** changes, the inputs to the temporary table processing system **32** are modified to update the information in the temporary tables **30** with new totals. In the preferred embodiment, the temporary table processing system **32** updates the data in the temporary tables **30** every 60 minutes, but the time interval can be set by the programmer. Depending on needs, the update is performed based on such factors as product turn over, product manufacture time, number of orders taken per interval, and so on. The update should be performed often enough so that users acquire real-time data based on these factors.

[0023] The preferred embodiment shown in **FIG. 1** contemplates a manufacturing facility that has a number of products in various stages of completion, or a re-manufacturing facility with the same traits. Such a facility may have a system as represented by **FIG. 1** merely to report the status to internal users. Additionally, an organization with these capabilities may want to make the information available to field sales personal, or even directly to potential customers.

[0024] **FIG. 2** is a representation of a preferred embodiment for the statistical prediction of future performance process. In general, a process can be measured for performance if certain specification limits are set to measure it. To do so, the system must determine how many times the process produces a defect, or an output, that does not conform to the specifications that were set, as compared to how many times it produces a non-defect, or a successful output, within a given number of opportunities. In this manner, it is possible statistically to predict the output of a process by sampling performance of the process and analyzing that data. The process can then be assessed and adjusted to improve future performance as against the specification limits.

[0025] Referring to **FIG. 2**, the flowchart provides a description of a preferred process for creating the statistical calculation to indicate quality. Overall, **FIG. 2** is divided into two separate parts. Part A describes updating the temporary tables **30** in the database **14**, and Part B describes the process flow of the statistical calculation.

[0026] Referring to **FIG. 2**, Part A, in more detail, upon initialization of the updating process **34**, data from the database **36** is obtained by a fetch order for information **38**. This is sometimes referred to as querying the database. In this embodiment, the data obtained is only for the orders that have occurred within the past year for statistical purposes. However, any time period can be chose based on design choice. The next step is where orders that have not been shipped are removed from consideration **40** because this portion of the system is only concerned with shipped orders. Next, the initial calculation **42** is performed which requires fetching the maximum ship date from that order at **44** and fetching the customer's requested date for delivery **46**. In this embodiment, each order may have more than one product, and the maximum ship date referred to in **44** is the date that the last product on the order was actually shipped. The customer requested date for delivery **46** is subtracted **48** from the maximum ship date **44**. At this point, the process adds **52** the time needed for shipping the product to the customer **50** to the difference between the maximum ship date and the customer's requested date for delivery providing the result to update the table **54**. Updating the information that is already in the table or providing a new entry then modifies the table. The process then checks whether the last order was processed **56**, and if not **56*a***, the process returns to the initial calculation step **42** where the previously-described process is repeated on the next order entry. If the order is the last **56*b***, then the process continues to the statistical calculation section of part B. It is important to note that the invention anticipates the use of any number of statistical calculations to predict the capability of a process,

both known and as yet unknown. The preferred embodiment uses a value known as the "Z" score to provide information about process capability.

[0027] To calculate the mean **58**, the data is added together and divided by the number of entries. In statistical equations the mean is customarily represented by the Greek letter mu (μ). The next act in the process is to calculate the variance **60**. As is well known in the art, this is done by subtracting the mean from each entry in the table to determine a deviation for each entry, squaring each deviation, summing the squared deviations and dividing that sum by the number of entries. The Greek letter sigma σ represents variance when shown with an exponent of 2 (i.e., σ^2) The next act is to calculate the standard deviation **62**. This calculation is also well known in the art, and it is equal to the square root of the variance. The standard deviation is traditionally represented by the Greek letter sigma (σ) with an exponent of 1. The next step is to determine the values for Z long-tern and Z short-term **64**. In general, Z-scores are well known in the art of statistics. Z long-term (Z_{LT}) is calculated from the standard deviation and the average output of the current process. Used with continuous data, Z_{LT} represents the overall process capability and can be used to determine the probability of out-of-specification parts within the current process. Usually process capability is measured in defective parts per million opportunities, or DPMO.

[0028] In the preferred embodiment, the Z score is determined by first setting an upper specification limit (USL) and a lower specification limit (LSL), also referred to as tolerance limits. These limits are designated either arbitrarily, or as a result of researching customer needs, to determine the goals or tolerance of the process. As a measure of quality, any data point that falls outside of the USL and LSL is then considered a defect. The Z long-term (Z_{LT}) value is then calculated by use of the formula:

$$Z_{LT} = \min\left[\frac{USL-\mu}{\sigma}, \frac{\mu-LSL}{\sigma}\right],$$

[0029] where USL is the preset upper specification limit, LSL is the preset lower specification limit, μ is the mean, and σ is the standard deviation. In the preferred embodiment, the minimum result of the two expressions is taken as the measure of performance for the process. The user may then interpret the result. Interpretation may include applying the result to a normal or other distribution to determine the percentage of defects produced with a given number of opportunities. It is contemplated and believed within the scope of the present invention that interpretation of the results to determine DPMO could be easily automated as well. Z values can also be calculated in other ways; for example, both expressions can be used in some calculations instead of using the minimum of the two.

[0030] Continuing with the preferred embodiment, the Z_{LT} value is then used to determine the Z short term (Z_{ST}) value by using the formula: $Z_{ST}=Z_{LT}+1.5$. This is an estimation of performance based upon the idea that the performance of a process will deteriorate over time. Thus the short-term performance represented by the Z score should be better than the long-term performance, and adding 1.5 to the long-term Z score estimates the short-term Z score. The Z scores are then moved to the update table **66** where they can be called for display. In the preferred embodiment, Z short term (Z_{ST}) is the standard scale for reporting performance quality based on a target goal of 6 Sigma (If $Z_{ST}=6$ then DPMO=3.4). At this point the process ends **68**.

[0031] Accordingly, the present invention includes a method for measuring product shipment process capability that includes maintaining a database having therein data fields indicative of at least an order, a maximum ship date, a customer requested data, and a product category for each order, and then fetching order information for all orders that have a valid maximum ship date. The method includes subtracting the customer requested date from the maximum ship date and producing a difference value therefrom. Next, a predetermined number of days is added to the difference value to provide a shipment quality metric for each order. The method next includes using the shipment quality metric in a statistical calculation to indicate process quality.

[0032] In a preferred embodiment, the order information fetched from the database is only for those orders that were placed within a given time period. The statistical calculation can include determining a value for an upper specification limit and a lower specification limit, and then displaying the percentage of times the shipment quality metric was greater than the upper specification limit and displaying the percentage of time the shipment quality metric was less than the lower specification limit. The invention can include setting a value for at least one specification limit and computing and displaying a statistical score based upon the specification limit and the shipment quality metrics where the statistical score is a measure of process capability. The process includes periodically repeating the fetching, subtracting, adding shipping days, and determining a statistical calculation at regular time intervals. The statistical calculation is preferably calculated and displayed for each product category. Further, the Z values can be displayed on a scale representing a range of Z values with an overlapping needle to indicate current performance as shown in **FIG. 7**.

[0033] The invention also includes a computer program implementing the aforementioned method. The computer program is stored on a computer readable medium and causes one or more computers to query a database that contains the information detailing orders, a requested delivery date, a maximum ship date, and a product category for a number of products. In processing the data, the computers are caused to ignore orders with no maximum ship date, and subtract the requested delivery date from the maximum ship date and add an adjustment value to obtain a shipment quality metric. The query, subtraction, and addition acts are repeated for the number of shipped products, and the computer program has instructions to process the shipment quality metrics to determine overall shipment quality.

[0034] Preferably, the shipment quality metrics are processed to provide a statistical measure of process capability and are regularly reprocessed by repeating the aforementioned acts at regular time intervals. The regular time intervals are dependent upon the system being evaluated and are preferably set such that the user of the system perceives that the shipment quality metrics are updated in real-time.

[0035] Preferably, the set of instructions in the computer program that process the shipment quality metrics includes instructions to determine a mean and a standard deviation of

the shipment quality metrics, and to designate an upper and lower specification limits, and then to determine a Z long-term value by subtracting the mean from the upper specification limit and dividing the result by the standard deviation. The Z long-term value is then displayed. Further instructions can includes an estimated value for Z short-term by adding a constant to the Z long-term value.

[0036] Another embodiment of the invention includes a computer data signal representing a sequence of instructions that, when executed by a processor, cause the processor to maintain a database of data having therein an order number, a promise date, a request date, a maximum ship date, and a product category for each product. The instructions in the computer data signal cause the processor to obtain the data from each order that has a valid maximum ship date and create an upper specification limit by adding a predetermined number of days just prior to a customer's requested delivery date and to create a lower specification limit by adding a predetermined number of days after a customer's requested delivery date. The processor is then caused to compute and display a statistical value providing an indication of process capability. These instructions are periodically repeated either at regular time intervals, in real-time, or on a on-demand basis that can include recalculating the statistical value each time a user requests the information. In order to calculate the statistical value, the computer data signal preferably has instructions to determine a mean value and a standard deviation and to subtract the mean value from the upper specification limit and divide a result by the standard deviation to create a first Z-value. The lower specification limit is subtracted from the mean value and the result is divided by the standard deviation to create a second Z-value. The minimum of the two values is then chosen as the statistical value.

[0037] The computer data signal can also have instructions to project what the number of defects would be, given one million opportunities. The defects per million (DPM) is based upon a normal distribution. Applying the Z_{LT} or Z_{ST} to a normal distribution table provides the predicted DPM. While the current preferred embodiment does not describe a method to do this electronically, it can easily be done by the use of a simple look-up table, well known in the art of computer programming. The Z_{LT} or Z_{ST} value is found on the table and the associated DPM value is obtained. The DPM value is then displayed as the number of defects per one million opportunities. The instructions in the computer data signal can also cause a processor to decide which of the first and second Z-values are a minimum value and then to display the minimum value identified as a Z long-term value. A constant can then be added to the Z long-term value to determine and display a Z short-term value.

[0038] **FIG. 3** is a flow diagram of the process of a preferred embodiment for displaying inventory availability. The process begins at the start step **70** and the first step is to fetch all saleable items in the inventory **72** from the database. The program then must determine whether each record contains a date indicative of whether the product is available to ship, referred to as an available to promise (ATP) date **74**. If a particular record does not have an ATP date **74***a*, the record is ignored **76**. If it does have an ATP date **74***b*, then the process next counts the number of days between a current date (i.e., typically today) and the ATP date at **78**. The resulting number is then provided to the next part of the process to determine which of the messages is appropriate to display for this record. If the number of days between the current date and the ATP date is greater than 500 days **80**, then the message "Call for Availability" is displayed **82**. If the number of days between the current date and the ATP date is greater than 2 days and less than or equal to 500 days **84**, then the message "Shipment within number where number is the number of days between the current date today and the ATP date days"**86** is displayed. On the other hand, if the number of days between the current date today and the ATP date is less than or equal to two **88**, then the message "Immediate Shipment" is displayed **90**. Once the appropriate message is displayed, the process ends for that entry at **92**.

[0039] **FIG. 4** is a flow diagram representing the work in progress (WIP) table update process for one preferred embodiment. The process updates the WIP table that is in the temporary tables of the database to ensure the data displayed is the most current available. The process begins after initialization **93** by fetching the booked orders that have been promised a shipping date **94**. The system then determines whether the record, already exists in the current WIP table **96**. If the record already exists **96***a*, the order number, order promise date, request date, and the product category are fetched **98**, and they are used to update the current entry in the WIP table **100**. This portion of the process is then complete at **101**. If the record does not exist **96***b*, the order number, order promise date, request date, and the product category are fetched **102**, and are used to create a new entry in the WIP table **104**. This portion of the process is then complete at **101**. The entire process repeats until all the entries for the WIP table have been either updated or created.

[0040] **FIG. 5** is a flow chart representing the alert setting and display process. The system provides for both proactive and reactive alerts, allowing use of the information to repair problems in the shipping process. The word "proactive" is meant to show that an alert may be set while there is still time to rectify a possible problem in the process, in this case, shipping. The proactive alert allows the process owners to make adjustments or take special action in order to avoid a late shipment. The reactive alert may not provide the same information, but the information is still useful to ensure that recognized problems do not occur.

[0041] Continuing with **FIG. 5**, the alert setting and display process begins after an initialization **105**, by fetching the WIP table **106** and then processing with each order in the WIP table. The program fetches the respective product category **108** and then the maximum promise date **110**. The maximum promise date is the latest date promised to the customer for shipment. The request date is then fetched **112**, and then the process determines whether the promise date is after the request date **114**. If the promise date is later than the request date **114***a*, then a "Promise Alert" is set and displayed for that order **116**, and the process moves to the order already shipped question **118**. If the maximum promise date is prior to the request date **114***b*, then the process skips the promise alert step **116** and moves directly to the order already shipped **118** determination. If the order has already been shipped **118***a*, the process moves directly to check to see if all the orders have been processed at **124**. If not **118***b*, the process then determines whether the request date is within a preset number of days from a current data **120**. In this embodiment, the preset number of days is two. If the

request date is not within two days of the current date **120***a* then the process moves to check if another order must be processed at **124**. If the request date is within two days of the current date **120***b*, then the "Ship Alert" is set and displayed **122**, and the process moves to check if all the orders have been processed **124**. The process then repeats for all orders **124**, **124***a*. Once all of the orders have been processed **124***b*, the alerts are accumulated and displayed for the individual orders sorted by product category and type of alert **126**. The alerts provide opportunities for management to fix problems before they happen.

[0042] **FIG. 6** is a representation of the process used to display the orders and revenue for a past period of time. The process is initiated automatically when the information is requested, or on a regular pre-programmed basis **127**. In a preferred embodiment, all of the order information, including revenue for each order, is retrieved from the database **128**. The program then captures all the orders **130** in a previous period of time, in this example, a last year. The Orders table is then updated **132** with the data from the capture **130**. At this point the orders in the table are sorted by month and category **134**. The last step provides that the orders, order totals, revenue and revenue totals are displayed in tabular format **136** for viewing in an easily understandable way. After all the orders are displayed the process is completed at **138**.

[0043] In one aspect of the invention, a method for reporting status of work in progress is disclosed wherein a database is periodically queried for information about product orders. The information obtained for each order includes at least an order number, a promise date, a request date, a shipment date, and a product category for a plurality of products/services offered. The method further provides that one step conducts comparing the promise dates and the request dates to determine the difference. When compared, it is decided whether an alert should be set, including setting a proactive promise alert if a promise date is later than a request date for a given order. The last step of the method displays the proactive promise alerts. This is formatted to make it easy to find the information needed. It can be displayed in table format with the order number, product category, and the type of alert.

[0044] In accordance with another aspect of the invention, a computer-readable medium is disclosed, having stored thereon one or more computer programs. The computer program(s), when executed by one or more computers, causes the one or more computers to display electronic shipment alerts. The program begins by instructing the one or more computers to populate a database with data with at least the following for each order: an order number, a promise date, a request date, a shipment date, and a product category. The one or more computers are additionally instructed to periodically query the database and compare promise dates to request dates. The timing for automatically querying the database can be adjusted as necessary to provide up-to-date information. When the promise date is compared to the request date, the one or more computers are instructed to set a proactive alert if the promise date is later than a request date, and set a reactive alert if the shipment date exists and the request date is less than a user-defined number of days prior to a current date. The one or more computers are then instructed to display any promise and shipment alerts by product category and type of alert.

[0045] In accordance with yet another aspect of the invention, a computer data signal is disclosed. The signal represents a sequence of instructions that, when executed by one of more processors, cause the one or more processors to display proactive and reactive alerts by product/service category and type of alert. The sequence of instructions first cause the one or more processors to populate a database with at least: an order date indicating a date an order is initially made, a request date indicating a date when a customer requests delivery of the order, a shipment date, when available, indicating a date when actual shipment will occur, and a product/service category for each order for a product/service. The one or more processors are further instructed to query the database and compare the promise date to the request date for each order, and to check for the entry of a shipment date for each order. The one or more processors are then instructed to set a proactive alert if any promise date is later than a request date, and to set a reactive alert if a shipment date exists for an order and the request date is less than a user-defined number of days prior to a current date. The one or more processes are lastly instructed to display all proactive and reactive alerts by product/service category and type of alert.

[0046] **FIG. 7** is a representation of the Z-value graphic indicator in accordance with another aspect of the invention. The preferred embodiment provides an indicator written in Java. The ranges of values on the scale change automatically to best reflect the value to be indicated, and the needle indicates the value of the current calculation.

[0047] Each of the processes just described can be used alone, or in conjunction with each other in various combinations. In one preferred embodiment, the reports are displayed on a number of web pages within an intranet system. It is automatically updated and the information is available 24 hours a day on an as-needed, on-demand basis.

[0048] The present invention has been described in terms of the preferred embodiment. While the preferred embodiment uses computers that are communicating through some form of a network, it is understood that other embodiments of the invention may involve the use of different technologies. It is recognized that equivalents, alternatives and modifications that are different from the preferred embodiment exist, and they are within the scope of the appending claims.

What is claimed is:

1. A method for reporting status of work in progress, comprising the steps of:

periodically querying a database that contains data indicating an order number, a promise date, a request date, a shipment date, and a product category for a plurality of products/services offered;

comparing the promise dates and the request dates;

setting a proactive promise alert if a promise date is later than a request date for a given order; and

displaying the proactive promise alerts with the order number by product category and type of alert.

2. The method of claim 1 further comprising the steps of:

setting a reactive shipment alert if the shipment date exists and the request date is less than a user-defined number of days prior to a current date; and

displaying any reactive shipment alerts with the order number together with the proactive promise alerts.

3. The method of claim 2 wherein the user-defined number of days is equivalent to a number of days required for shipping a product to a customer.

4. The method of claim 1 wherein the querying of the database is conducted automatically at regular time intervals.

5. The method of claim 1 wherein the steps of the method are repeated automatically in real time.

6. The method of claim 1 further comprising repeating the steps of the method every time a request for information is made.

7. The method of claim 2 wherein the proactive promise alert allows for correction of a potential late shipment and the reactive shipment alert provides data to prevent future late shipments.

8. The method of claim 1 further comprising the steps of reacting to a proactive alert by performing one of:

modifying the promise date to coincide with the request date; and

notifying a customer that the request date cannot be fulfilled as desired.

9. A computer-readable medium having stored thereon one or more computer programs that, when executed by one or more computers, causes the one or more computers to:

populate a database with data to include an order number, a promise date, a request date, a shipment date, and a product category for a plurality of orders;

periodically query the database and compare promise dates to request dates;

set a proactive alert if the promise date is later than a request date;

set a reactive alert if the shipment date exists and the request date is less than a user-defined number of days prior to a current date; and

display any promise and shipment alerts by product category and type of alert.

10. The computer-readable medium of claim 9 wherein the user-defined number of days is equivalent to a number of days required for shipping a product to a customer or providing a service to a customer.

11. The computer-readable medium of claim 9 wherein the query of the database is conducted automatically at regular time intervals.

13. The computer-readable medium of claim 9 wherein the one or more computer programs cause the one or more computers to repeat the actions of claim 9 every time a request for information is made.

14. The computer-readable medium of claim 11 wherein the regular time interval is between 0 and 60 seconds.

15. The computer-readable medium of claim 11 wherein the regular time interval is greater than 1 minute.

16. A computer data signal representing a sequence of instructions that, when executed by one of more processors, cause the one or more processors to:

populate a database with an order date indicating a date an order is initially made, a request date indicating a date when a customer requests delivery of the order, a shipment date, when available, indicating a date when actual shipment will occur and a product/service category for each order for a product/service;

query the database and compare promise dates to request dates for each order and check for the entry of a shipment date for each order;

set a proactive alert if any promise date is later than a request date;

set a reactive alert if a shipment date exists for an order and the request date is less than a user-defined number of days prior to a current date; and

display all proactive and reactive alerts by product/service category and type of alert.

17. The computer data signal of claim 15 wherein the user-defined number of days is equivalent to a number of days required for shipping a product/service to a customer.

18. The computer data signal of claim 15 wherein the query of the database is conducted automatically at regular time intervals.

19. The computer data signal of claim 15 wherein the computer data signal causes the one or more processors to repeat the actions of claim 15 every time a request for information is made.

20. The computer data signal of claim 17 wherein the regular time interval is between 0 and 60 seconds.

21. The computer data signal of claim 17 wherein the regular time interval is greater than 1 minute.

22. The computer data signal of claim 15 wherein the computer data signal causes the one or more processors to allow user modification of the promise date to coincide with the request date in response to the proactive alert if the product/service is available by the request date.

* * * * *